Virtual Reality Games for Upper Limb Rehabilitation

Second Semester Report
Spring Semester 2014

- Full report -

by
Joshua Wallace
Ryan Nash
Jacob Thornton

Prepared to partially fulfill the requirements for
ECE402

Department of Electrical and Computer Engineering
Colorado State University
Fort Collins, Colorado 80523

Project advisor: ___Sudeep Pasricha___

Approved by: _____Sudeep Pasricha___

ABSTRACT

Numerous people go through upper limb rehabilitation every year due to strokes, accidents or surgery. The goal of our project is to provide a rehabilitation environment that encourages and stimulates the patient in way clinical rehab currently does not. Our games focus on stimulating two types of motion; lateral and extension. Through the use of simple games we can disguise these motions as the controls to our games. We have developed a number of virtual reality games which are aimed towards assisting patients in upper limb rehabilitation. Our games will be developed on the Leap Motion Controller interface device and will be played on a PC. We have developed our games using HTML5 and JavaScript along with various game development libraries. By making our games web-based the player will have the ability to simply log in to a website from their home computer to access and play the games. This also enables the collection and consolidation of game data for each patient which can then be viewed by the patient's physical therapist in order to track progress. This system would be useful for many people across the globe.

This project has been a senior design project for two years at Colorado State University. The first version of the project used a Web Cam to implement Object Recognition and user tracking. The second version of the project attempted to do the same thing on a mobile computing platform. We took the successes and failures of these projects into account for our own purposes.

We went through a number of phases to reach our current solution. Initially we started developing with the Microsoft Kinect. This research consisted of investigation into available Kinect libraries, as well as supported functionality. We needed to get an idea of the type of games that the Kinect could support. The most useful part of the Kinect was the skeleton class. It allowed the programmer easy access to information about the placement of extremities. This data could then be monitored to catalogue the user performing prescribed events.

The goal of our project, however, was not to monitor the entire user. Instead we were tasked with the rehabilitation of upper limbs. This skeleton provides us with extra, unnecessary features. Fearing these extra features were just something to go wrong, we sought a system that isolated the upper body. The Leap Motion Controller was released over the summer and uses infrared cameras to report high resolution information about the user. Using this device we could monitor the number of fingers extended, whether the users hand was balled in a fist or flat, and most importantly the speed and direction of the movement.

Finally we had to keep in mind the state of our users. The Augmented Reality Games system is intended for people suffering from upper limb rehabilitation, and many of these people struggle with simple tasks. With this in mind, we built our games to be played with the user resting his hand on the table, while the Leap Motion Controller was mounted above. This restricted motion to the X and Z dimensions, but benefited users who otherwise could not extend their arm in front of them.

Our efforts have led to a one of a kind setup that allows patients to continue rehab in the comfort of their own home. Unlike rehabilitation done without this technology, progress can be tracked by storing results on a server. These results can be viewed by the patient and therapist to get an idea of the progress that has been made. The end result of our system is a new way of doing in home rehabilitation that offers many of the benefits of rehab performed on site.

Next semester our project will be handed off to another group of students who will continue and enhance our work. These students will look to make additions of their own, which could vary from implementing new hardware to designing games built for full body rehabilitation. Another area of interest would be the inclusion of biometric sensors, a type of hardware that seeks to provide insight into muscle activation and usage. This data could be used to provide the therapist with information about what muscles are being used when a patient is playing a game.

Another important part of next year's project would be support for the graduate students performing studies with the system. They would be there to answer questions about how to use the website, assist them with setting up new users and provide support and fixes for any bugs found.

We would also like to make a few recommendations of our own. The first is the implementation of an assignments database. With this feature the therapist can assign his patients to play any combination of games he/she has already configured. On assignment and completion the therapist and patient would be emailed with the address in the database.

The second improvement would be to implement a preferred hand. Using the leap you can identify the left and right hand. If a user choose to place both hand in the playing area for comfort (this was rather common), we would like to isolate the correct hand by using a setting from the database.

The third would be to extend the project for users who are further along in rehab. Currently the games are designed for people who struggle with simple motions. We made the conscious decision to limit the area of gameplay to two dimensions. The Leap allows us to take advantage of three degrees of motion. We suggest building games for more advance users that require a third degree of motion.

TABLE OF CONTENTS

i	Title	
ii	Abstract	
iii	Table of Contents	
iv	List of Figures and Tables	
7	Introduction	I.
10	Summary of Work	II.
10	Section 1 - Hardware	
12	Section 2 - Software Selection	
13	Section 3 – The Database	
14	Section 4 – The Web Interface	
21	Section 5 – The Game Engine	
24	Section 6 – Meteors Game	
25	Section 7 – Pirates Cove	
27	Section 8 – Maze Game	
29	Section 9 – Water Drop Game	
30	Section 10 – Whack-a-Mole	
31	Section 11 – Database Improvements	
34	Alternatives, Testing and Ethics	III.
37	Clinical Trials	IV.
41	Conclusions and Future Work	V.

43 References

A-1 Appendix A - Abbreviations

B-1 Appendix B - Budget

C-1 Appendix C – Additional Documents

D-1 Appendix D – User Manual

E-1 Appendix E – Meteors Game

F-1 Appendix F – Pirates Cove

G-1 Appendix G – Maze Game

H-1 Appendix H – Water Drop

I-1 Appendix I – Whack-a-Mole

LIST OF TABLES AND FIGURES

Figure 1: System Setup
Figure 2: Database Model 13
Figure 3: Default Layout View 14
Figure 4: Login View 15
Figure 5: Register View 15
Figure 6: Default View 16
Figure 7: Main Page View 17
Figure 8: Ribbon 18
Figure 9: Information Window 18
Figure 10: Profile View 19
Figure 11: Progress View 19
Figure 12: Game Engine 20
Figure 13: Meteor Game 24
Figure 14: Pirates Cove 25
Figure 15: Maze Game 27
Figure 16: Water Drop 29
Figure 17: Whack-a-Mole 30
Figure 18: Craig playing Whack-a-Mole 37
Figure 19: Craig's Results 38
Figure 20: Debbie playing Pirates Cove 39
Figure 21: Debbie's Results 40

Table 1: Team Expenditures 32
Table 2: Personal Expenditures 32

Chapter I - Introduction

Physiatry, commonly known as rehabilitation, dates back to the two major wars of the 20th century. Physiatrists recognized the need for physical stimulation of soldiers before they could return to combat or everyday life. Major Frank Granger was designated the head of this new physiotherapy service, setting up reconstruction units in fifty-three army and general hospitals across the nation. Patients were assigned exercises of varying difficulty based on the progress of their rehabilitation. From these seeds the profession of physical therapy grew into its own field. Specialists emerged to usher the field into a respected status.

Physical therapy grew in the thirties to include elements like hydro and spa therapy. A Society for Physical Therapists was founded in 1938. World War II saw an explosion in the field. The Baruch Committee was founded to explore advancements in the field. The committee focused on education, teaching, research, public relations, rehabilitation, hydrology, occupational therapy and prevention and body mechanics. This council provided funds to universities, backed by Bernard Baruch, to advance the field. The end of the war saw the emergence of the Archives of Physical Medicine and Rehabilitation, the premier journal in the field. During the war Physical Therapists and Rehabilitation Experts began to differ. Physical Therapists (Physiatrists) are people who have graduated medical school and coordinate services for people who are in need of rehabilitation. Rehabilitation specialists administered those treatments, focusing on speech, occupation, corrective and recreational therapy[1].

Millions of Americans are effected by traumatic brain injury (TBI), stroke or cerebral palsy (CP). These impairments result in restricted movement which makes everyday activities difficult. We developed a system with the intent of using augmented reality to rehabilitate patients with neurological and physical impairments.

Our current system has been designed to focus on the physical elements of therapy, especially occupational therapy. Our task from the beginning was to incorporate elements of everyday life into our apparatus, thus the concept of Augmented Reality games. The concept behind rehabilitation has been to help people assimilate into society despite injuries suffered in the workforce, playing sports or medical misfortune like strokes or seizures. With this in mind, we must design games to encourage users to replicate these actions.

These actions can range from the simple, like holding a ball or a cup, to the complex, like completing a puzzle. Our system must cover an entire spectrum of tasks, and prompt users to perform them at varying levels of difficulty. This can be done in a number of ways. The first is to program varying levels of difficulty into the game through time limits, speed of game play and range of motion variables. The second is by changing the object the patient must use. By having the patient use varying size and weight combinations we can increase the difficulty of the motions.

The other end of the problem is tracking the progress of a user and managing this data. Because the therapist is not there in the room with the patient, we need a way of chronicling and measuring the actions of the user. To do this we created a database that tracked users with a login system. This data can then be accessed by the patient and corresponding therapist to track the progress that has been made over the course of treatment.

It should be noted that this system is not meant to take the place of a full rehab regimen. When designing this system we must take into consideration the patient's ability to cheat the system. When certain muscles become unavailable to our body it compensates by using muscles not originally intended for the desired movement. Proper rehab should avoid this scenario. We have designed this system to supplement visits to a rehab expert. In office visits are checkpoints to ensure the patient has been performing the exercises correctly.

Our system simply disguises these repetitive actions in the form of simple games. Asking a patient to extend their arm while holding a weight can be disguised as collecting coins. Instructing a patient to perform lateral movements can be masked by having a patient collect water droplets onscreen. In this way were are attempting to inject some fun into the process of rehabilitation.

Throughout the process we attempted to keep in mind the future of the project. Our initial attempts at games have been simplistic. Most of the games focus on collecting objects or repeating patterns in various planes of motion. We don't want our system to be limited to these actions. The website has been written to allow for the inclusion of new games down the road. The server has been configured for an adjustable number of users. The games are resizable for different window sizes. The release of Windows 8 requires us to design games that will work for both Windows 8 and Windows 7.

We also cannot be guaranteed the processing power of the host computer. The Leap Motion controller requires that the computer has an AMD Phenom II or Intel Core i3, along with 2GB of RAM. We have no guarantees that patients have a host computer that meets these specifications. As a result we must do everything we can to avoid stressing the computers resources. This means simplistic images, limiting the number of particles onscreen and developing efficient code.

Designing a good product requires keeping all of this in mind. To do that we needed an efficient division of labor and good communication to ensure all goals were being met. We used sites like drop box, Bitbucket and Google drive to enhance the sharing of information. We had to keep access to these sites restricted to ensure no leakage of information.

Summary of Previous Work

There have been two previous iterations of this work. Both have focused on the integration of real world objects into a computerized system for the in home rehabilitation of patients. The benefits of this type of system are huge. It reduces the strain on a physical therapists schedule, increases the quality of care for low mobility patients and reduces the costs of care.

The first iteration used a webcam with object recognition to track the movement of objects across the plane of vision. The students were able to successfully design a system the allowed the user to use their arm to initiate game play. Armed with the knowledge that the idea had legitimacy to it, our advisor decided to take it mobile.

The second iteration of the project attempted to implement a similar system on mobile devices, such as a tablet. This ended up more difficult than originally thought. Sate of the art mobile systems did not have the processing power required to handle games of this nature. Having learned this, the project had to shift back to platform based computing.

The first few iterations set the stage for our project. We needed a system that could handle the load object recognition demanded and allowed for increased quality of game play. Systems considered had a background in gameplay, specifically gameplay with the user's body as the controller.

Chapter II – Summary of Work

Section 1) Hardware

In order to fit the needs of upper limb rehabilitation we needed to investigate and select the best hardware that would fit the needs of our application. The selection of our hardware depended on accuracy, size, and ease of use. Our professor, Sudeep Pasricha, was kind enough to narrow down our options to either the Microsoft Kinect or the Leap Motion Controller. During the first few weeks of development we investigated the capabilities of each device and ultimately chose which device would best fit our needs.

Both the Microsoft Kinect and Leap Motion Controller make use of an infrared grid detection scheme to track bodily movement. The devices emit an infrared grid which is then detected by a series of infrared cameras. The data collected by the infrared cameras is sent through an image processing system which defines the locations of various parts of the human body. Although both of these devices make use of similar tracking technologies, they differ in several significant manners.

First, there may be a time when a patient must take the hardware home with them in order to fulfill their physical therapy outside of the facility. In this case, the smallest piece of hardware will be most desirable. The Microsoft Kinect measures roughly 11" wide, 3" deep, and 3" tall. This is considerably larger than the Leap Motion Controller which measures only 3" wide, 1.2 inches deep, and 0.5" tall. Given these measurements it is clear that the Leap Motion Controller is the most portable option.

Second, the ease of setup of each device must be considered. The Microsoft Kinect requires an external DC power supply along with a special USB cable. Along with these requirements, the Microsoft Kinect must be mounted in such a way that it has a view of the entire human body. However, the Leap Motion Controller requires only a single standard USB 3.0 cable to both power and provide communication from the device. It also only needs enough space to view a user's hand. Again, given these facts the Leap Motion Controller is clearly the best device for our application.

Finally, the accuracy of each device must be considered in order to provide the best feedback for the patients. According to Gelman et al. the Microsoft Kinect has an average resolution of 29 mm[2]. However, according to developers, the Leap Motion Controller has an accuracy of $1/100^{th}$ of a mm, or 10μm of resolution. Given this data it is obvious that the Leap Motion Controller is considerably more accurate than the Microsoft Kinect.

During our decision in choosing the best device for our application we considered several criterions. The first criteria we investigated was in regards to size, which we found that the Leap Motion Controller was significantly smaller than the Microsoft Kinect. The second criterion was in regards to ease of use, which we found the Leap Motion Controller to require less setup and space than the Microsoft Kinect. Our final criterion was accuracy, which we found the Leap Motion Controller to be significantly more accurate than the Microsoft Kinect. Considering all

three of these criterions it is easy to see that the Leap Motion Controller is the best choice for our application.

The next step in our investigation was to choose the best mounting position for the Leap Motion Controller. According to our advisor, the patients that will be making use of our device will need to rest their hands on a surface. This means that the user will be unable to hold their hand above the Leap Motion Controller which was the original use case for the device. In order to combat this we needed to design a simple solution that would allow the user to rest their hand on a table while still being able to be tracked by the Leap Motion Controller.

Mounting the device above a "playing area" proved to be the easiest portion of this task. The most complicated portion of this task was figuring out at what height the Leap Motion Controller needed to be mounted at and which surface would provide the best tracking capabilities. We began by simply taping the leap motion controller to a paint stir stick and taping the stir stick to various objects of varying height. Next, we used various surfaces for the playing field such as printer paper, different colored tables, and different colored gloves. During our investigation we found that with the Leap Motion Controller mounted roughly 12" to 18" above a black desk mat provided the most accurate and consistent performance.

The mount that was created was very rudimentary. It consisted of a vertical piece of wood that was cut to be about 18" tall and a horizontal piece of wood that was cut in such a way that would allow the Leap Motion Controller to be centered above the desk mat. Jacob had a friend design and 3D print a piece of mounting hardware that allowed the Leap Motion Controller to be easily attached to the horizontal beam of the mount and also allowed it to be adjusted in regards to depth in order to center the Leap Motion Controller above the playing area.

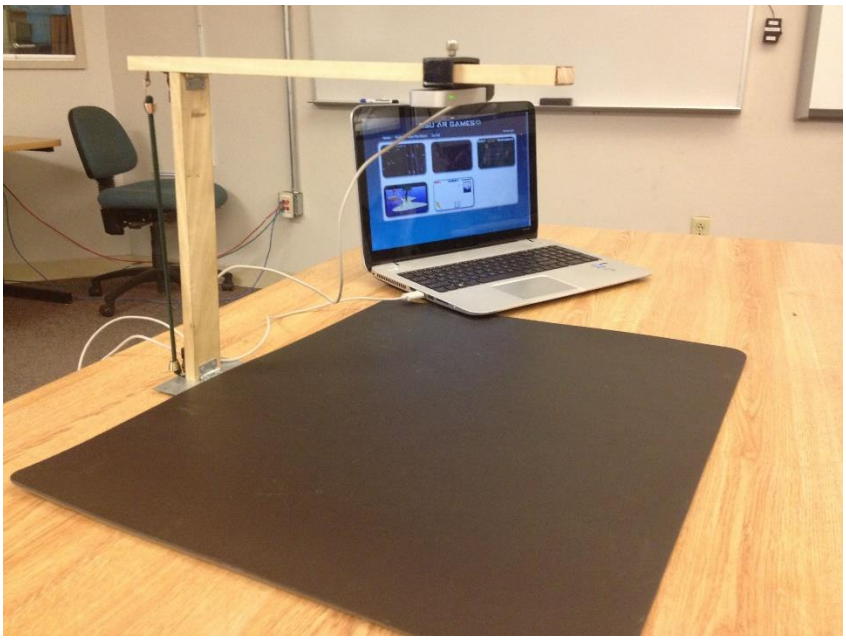

Figure 1) System Setup

Section 2) Software Selection

We wanted our games to be portable between various types of platforms such as PC, Mac, and Linux. In order to meet this requirement the programming language needed to be interpretive. This means that the code does not need to be compiled before it can run, it is simply interpreted at runtime. This cuts down on the need to compile the code numerous times for various different platforms. Given this requirement we narrowed down our programming language options to either Python or Javascript. Our final decision was made when we considered the requirements of each language. In order for us to use Python we would need to ensure that the end user had a compatible version of Python installed on their system. Also, Python would require us to manually integrate the Leap Motion API and any other development libraries. On the other hand, Javascript does not require any special installation or dependency integration on the users end. Its only requirement is that the user has an internet connection and a web browser. Given these facts we chose Javascript as our primary game development language.

In order to communicate with the Leap Motion Controller through a web browser we needed to include the Leap.js library. The Leap.js library provides us with numerous functions to retrieve desired data from the leap motion controller in real time. Once communication was established with the Leap Motion Controller we set our focus on researching the numerous Javascript game development libraries. We restricted our search to libraries that were open source and had decent documentation. After some time we came across the CreateJS game development library suite. CreateJS has numerous libraries that allow us to draw and manipulate images in the playing area, add sounds and background music, and create smooth animations and transitions. Now that our development environment was chosen we were able to continue with other software design choices.

One of the major benefits of making our games web based is that we can host all of them on a central web server. This web server could host a site that would allow patients to log in to play the games, and allow the therapists to track the progress of their patients. In order to make a powerful and easy to maintain website we needed to choose a suitable backend framework. For this task we chose the web2py framework for its abilities to easily maintain a database and deploy new updates and games. Given our choice of framework we could then begin designing the database structure and the website layout.

One of the major requests the occupational therapy group made was to include visual feedback to the user on his or her performance. One of the ways they suggested this was a visual graphic of performance. Highcharts JS is a plugin that allows for interactive representations of datasets. Highcharts supports many types of graphs, including Pie charts, scatter plots and line graphs, giving us a multitude of options for data representation. This plugin is implemented under the Game History tab, with the option for table viewing still available.

Section 3) The Database

We wanted our system to be able to keep track of the various patient and therapist information. In order to store this information we created a *person* table in our database. The *person* table contains the following fields: *first_name, last_name, email, username,* and *password.* When creating the *person* table we put in various requirements to ensure that the email provided was valid and unique along with ensuring that the username chosen by each person was unique. This would ensure that a user could not accidentally log into an account of another user who had the same username or email address. Also, whenever a new entry is added to the *person* table a unique ID is generated for this entry. This further ensures uniqueness between entries in the *person* table.

Now that we are able to store information regarding a *person* we needed to be able to identify which *person*'s are patients and which are therapists. For this task we created a *patients* table and a *therapists* table. The *patients* table contains a *patient_id* field which takes the unique ID of a *person* that is to be the *patient*. It also contains a *therapist_id* field that takes the unique ID of a *therapist* who is assigned to the *patient*. The *therapists* table contains a single *therapist* ID field that is the unique ID of the *person* that is to be a *therapist*. In order to retrieve the *therapist* that is assigned to a *patient*, we simply query the *patients* table with the *patients* ID. However, in order to find all *patients* assigned to a *therapist* we must query the *patients* table with the ID of the *therapist*. This is known as a one-to-many relationship, i.e. one *therapist* to many *patients*.

The final task that our system needed to do was track the progress of each patient. To do this, we created a *games_played* table which contains the following fields: *player_id, game_title, start_time, end_time,* and *score*. Whenever a new entry is created in the *games_played* table, the unique ID of the *patient* is stored in the *player_id* field. Whenever a therapist needs to retrieve progress information pertaining to a particular patient, they would simply query the *games_played* table with the unique ID of the *patient*.

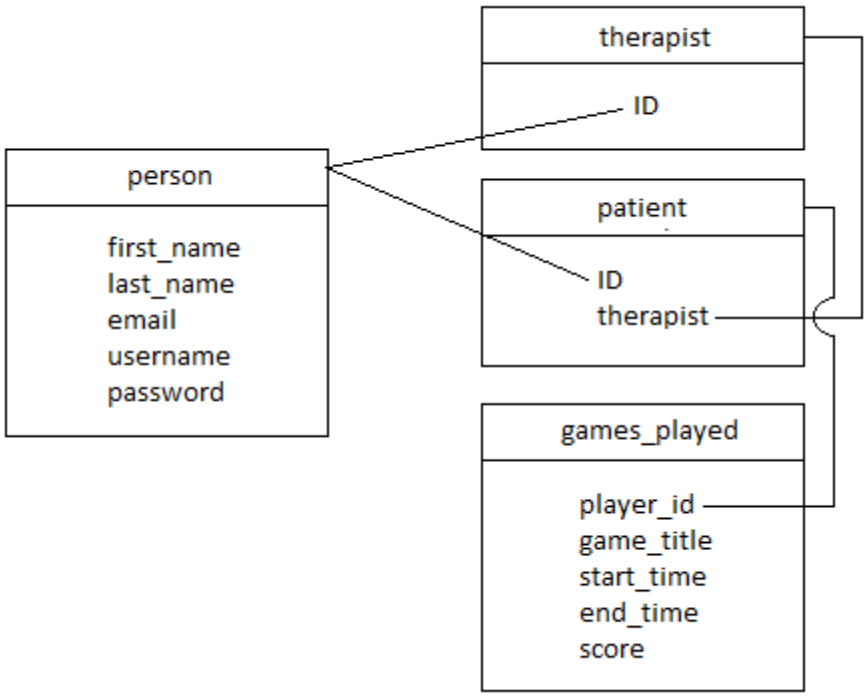

Figure 2: Database model

Section 4) The Web Interface

In order to make our system easy to use we developed a clean looking and intuitive web interface. The web interface needed to be able to allow users to log in, register, launch and play games, review progress, and change account information. We began by creating a default layout view which consisted of nothing but the webpage background and our team's logo.

Figure 3: Default layout view

We then created the login view which consisted of login form and the proper styling for its container. The login view inherited the default view which results in the page below.

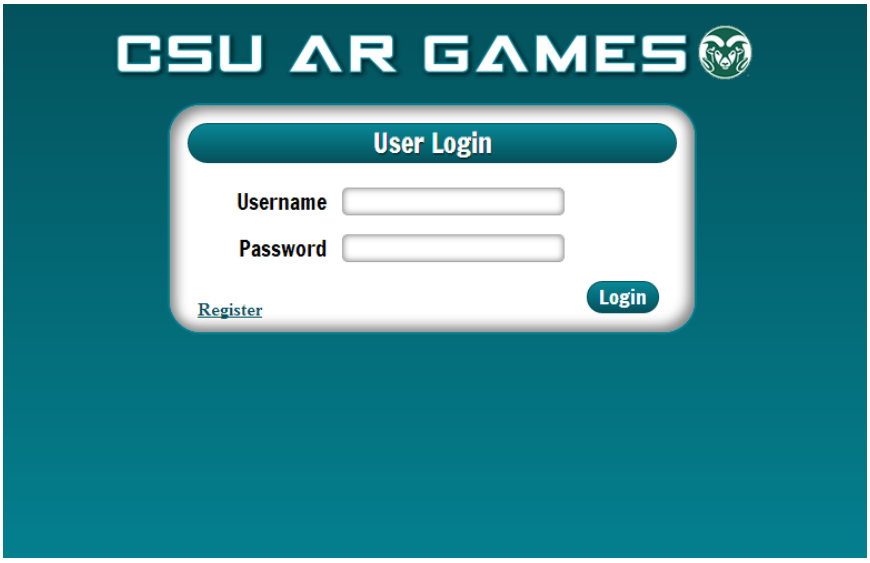

Figure 4: Login view

Next, we created the user registration view. This view consisted of a form that was dynamically generated depending on the various fields found in the *person* database table. By dynamically generating the form, we can ensure that the fields available to the user are always up to date. For example, if we were to add a birthday field to the *person* table, a *birthday* input would then be automatically rendered in the register form. The register form also inherits from the default view and is depicted below.

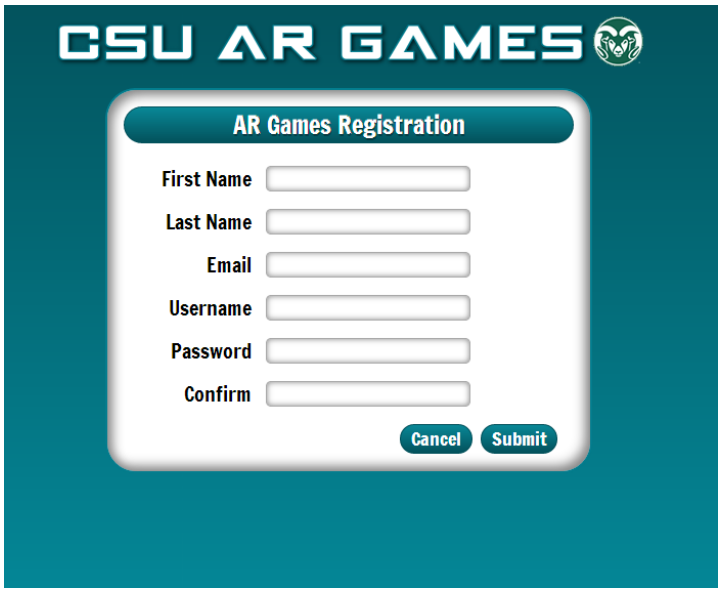

Figure 5: Register view

Once users were able to register and log in, we began developing the main web page. Our first task was to develop a main page view that would inherit from the default view. The main

page view consisted of the navigation tabs, user welcome message, and the copyright container in the bottom of the page. The main page view is presented below.

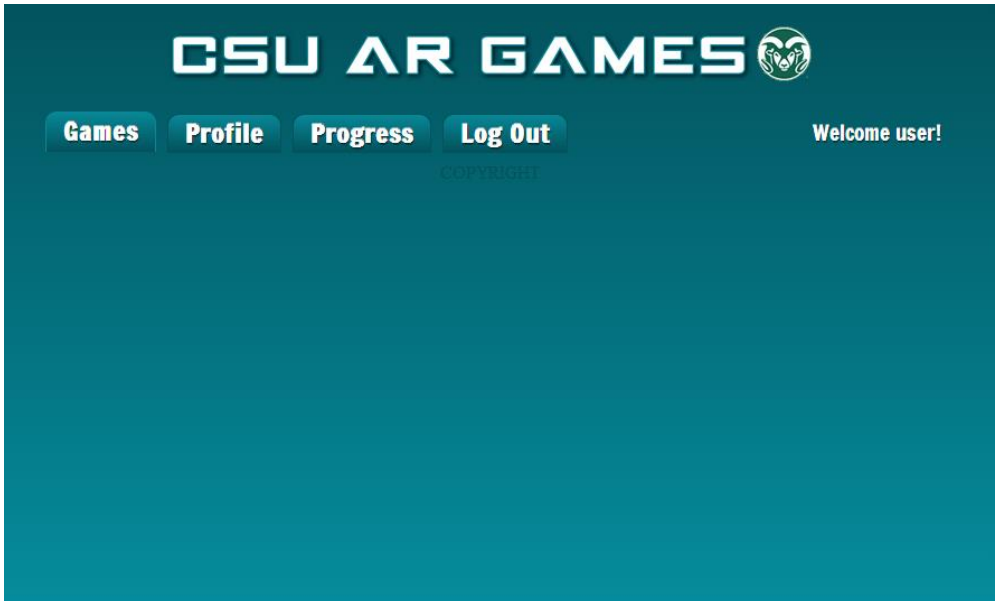

Figure 6: Default View

We needed to present the various games available to the user in an intuitive and clean manner. To do so we developed a sort of grid system that is similar to the ones found in many of the mobile device app markets. Also, the game options needed to be viewable on various sized displays. To meet this requirement we chose to implement a web design method known as "responsive web design". This means that whenever the user resizes the browser window or views the webpage on different sized displays the webpage will "respond" by dynamically resizing, reordering, or restyling the various elements so that everything will fit in the viewing area and still look clean. For example, on a large display our grid would be three columns wide, but on smaller screens the grid would dynamically reduce to two or even one column wide. Our game options view inherited the main page view and is presented below when the browser window was resized to force a two column grid.

Figure 7: Main page view

Whenever a user hovers over a game option, a ribbon will appear over the top of the game option. This ribbon contains the title of the game, an information button, and a launch game button. By clicking on the information button, an information window slides down from below the ribbon. In this information window we provide the user with a brief overview of the game such as what the goal of the game is, how it is played, and how points are awarded or revoked. Also, we inform the user of the type of motion that this game focuses on such as left and right, forward and back, up and down, or a combination of all three. This will allow the user to pick a game that will best fit their rehabilitation needs. Below we present the ribbon that appears above a game option and the information window that can be viewed.

Figure 8: Ribbon that appears when a user hovers over a game option

Score: 33
Maze
Goal: The goal of Maze is to guide the red ball through the designated path without colliding with the any walls. Move your hand left and right to to move the ball accordingly. Move your hand forwards and backwards to move the ball up and down. Points are given for each collision with a wall. Like golf, the goal of Maze is to acquire the fewest points possible.
Movements: Left and right. Forwards and backwards.

Figure 9: Information window that slides down from below the ribbon

Once our game option interface was created we created the user profile view. This view would allow the user to modify their profile information such as their email address or password. Whenever a patient views their profile they are also presented with their therapists contact information which is not modifiable by the patient. This feature was added because we thought that there might be a time when a patient needed to contact their therapist, but had lost their contact information. When the user selects the profile tab, the *person* database is queried to retrieve the patients information which is then used to render the patient information form. The *patients* database is then queried in order to retrieve the unique ID of the patients therapist which

is then used to query the *person* database in order to retrieve the therapists information for the therapist information form. The profile view inherits from the main page view and is presented below.

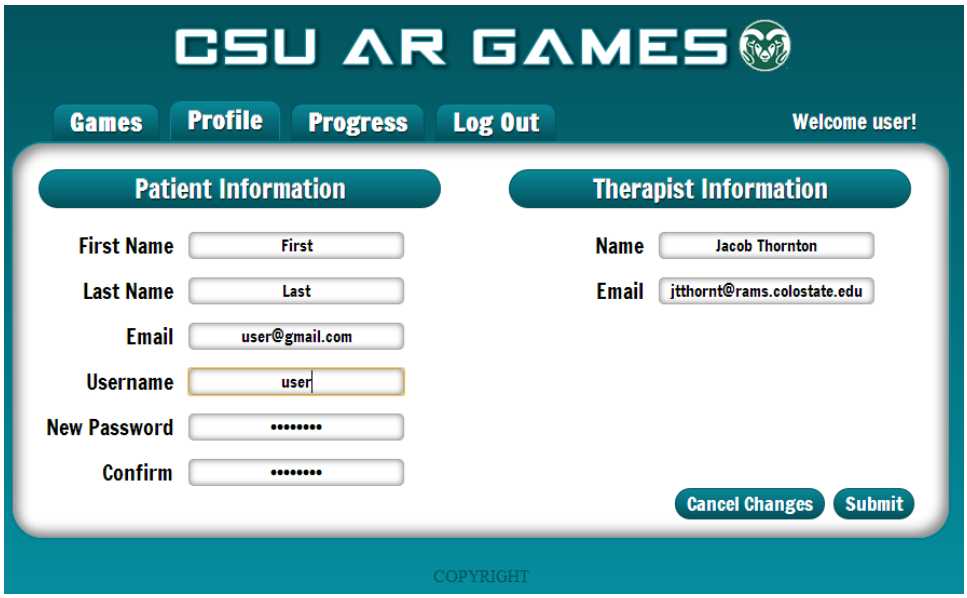

Figure 10: Profile view

The final view that needed to be created was the progress view. This view would present various gameplay information pertaining to the games that the patient has played. Currently this information is presented in a table, but we will be implementing software that would place this information in an interactive plot so that the patient/therapist can better visualize progress over time. When the user selects the progress tab, the *games_played* database table is queried with the unique ID of the *patient.* The data is then formatted and placed in a table for presentation. The progress view inherits from the main page view and is presented below.

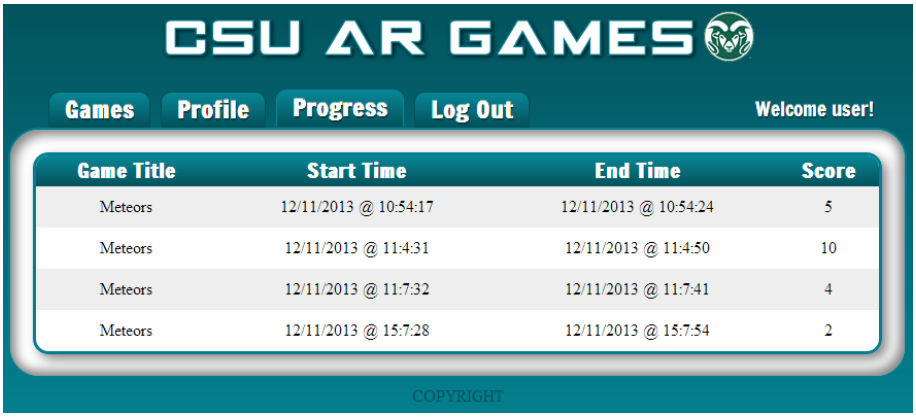

Figure 11: Progress view

Section 5) Game Design

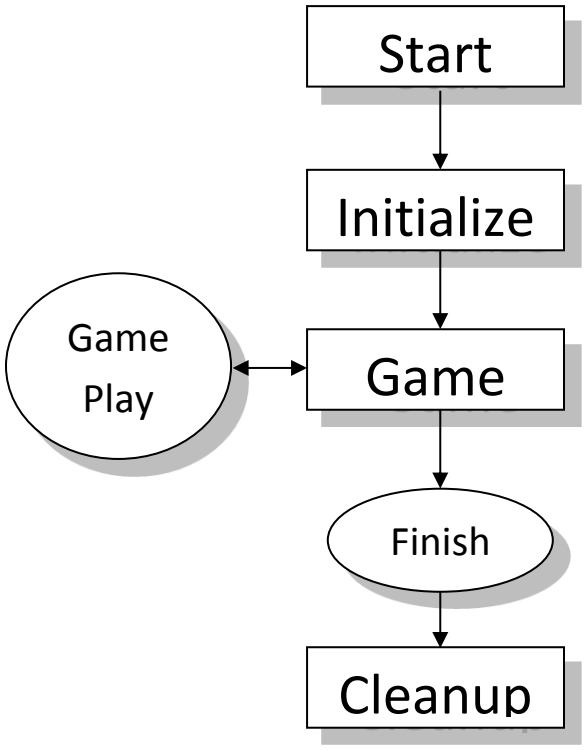

Figure 12: Game Engine

Game Flow

At the end of the first semester every game took input from a menu presented when the game was loaded. This menu works well for able bodied users, but for impaired users selecting the correct combination of inputs might prove difficult. Furthermore, we wanted to enable the therapist to define the gameplay from a remote location and keep it transparent from the user. In this way the game could be configured for each individual user to meet their needs.

To support this feature it required work and communication on both sides, from the website and the game itself. When the game loads it queries the website for a JSON object that contains the configuration details. This information includes time of gameplay, difficulty, sensitivity and preferred type of movement.

The time limit allows the therapist to configure the session to last for a specified time. After the specified time the game completes and reports the score back to the database. After conferring with the Occupational Therapy lab on campus we decided it was better to play for a given time rather than to a score goal. Playing to a score limit can take a variable amount of time, and the goal is to perform the exercises for an extended amount of time.

The sensitivity setting adjusts the correlation between movements with the users hand and movements on screen. A higher sensitivity input will result in a larger movement on-screen.

Using this input variable we can configure for patient range-of-motion. Range of motion is a therapy term that refers to how much movement a patient has. For patients with lower ranges of motion we can increase the sensitivity to help them play the game.

The pattern allow the therapists to rehab a desired type of motion. For games with two dimensional motion we have implemented seven patterns along with a random mode if feasible. The patterns are Left/Right, Up/Down, Snake, Circle, Square, Star, and X. If available the therapist can select between these patterns to rehab or check the progress of a certain type of motion. These pattern can be expanded upon to meet the needs of the therapists.

The difficulty has three options; easy, medium and hard. These levels adjust different parameters in each game, but have the overall intention of making the games easier or harder. For example, in the mole game the difficulty adjusts how long the mole is above the surface. In the treasure hunter game the difficulty adjusts the collision detection radius, making it easier or harder to collect coins.

Following the conclusion of gameplay after the specified time limit, the game and database must communicate again to store the results. This action is essential to track performance (and ideally progress) over periods of time. To do this we pass a string back to the parent (the site that calls the game script). In this string we include important information that can tell the therapist more about the session, like the difficulty, the final score and the duration of gameplay.

Just Right Challenge

In the second semester we had the opportunity to begin working with the CSU occupational therapy group. Communicating with them gave us insight into the specific needs we had to meet. One of these was the just right challenge. The configurability we build into our games revolve around this concept. This concept is to challenge the patient so they do not get bored, but not too much to prevent them from getting frustrated.

The way in which we altered the difficulty had to be done intelligently as well. For the games where the goal was to collect something, our initial idea was to resize the objects based on the difficulty. The harder the game the smaller the object would be. But after discussing this with Matt, our OT contact, this solution was undesirable because of Hemispatial awareness. Hemispatial awareness is a phenomena that occurs following a stroke where patients tend to ignore the senses on one side of their body. Because of this, we didn't want to confuse the senses by altering the sizes of the objects themselves, but rather the radius at which collisions were detected. In this way we can adjust the difficulty without confusing the senses.

The difficulty can be further increased by adding a feature where the pattern advances after a varying number of seconds. Currently objects are placed onscreen (Treasure Game and Meteors) and do not move until they are collected. A time interval can be added to advance the pattern if the coin is not collected within the allocated time.

Therapists have attempted to use games as a form of rehabilitation since they began to appear on the market twenty years ago. But every generation of games have proven too difficult to use for rehabilitation purposes. For example, a lot of excitement was generated within the rehab community by the Wii. The Wii featured a lot of fitness oriented virtual reality games. But these games proved too difficult, frustrating the patient.

Matt, our main contact in the OT department, harped on this concept. We feel that through the input settings, we offer this configurability. In testing with Matt, the range of difficulty for each game still needs to be fine-tuned. The most notable offender was the water drop game. The easy setting was far too easy for both of the patients who used our system. We anticipated these adjustments would be necessary because of our lack of experience designing games for rehabilitation purposes.

Scoring/Performance Metrics

Currently the only measure of performance we have is an accumulated score. This score differs per game as well as based on the difficulty setting of each game play. In other words, it might be easier to accrue one hundred points in the coin game versus the mole game. This variability can be worked around by viewing a score in relation to previous scores from that game only. But it would be better to have a consistent scoring metric across all games.

A slight improvement would be to report score per time unit played. This would remove the variability of differing time lengths. Rather than reporting a score of 1000 for ten minutes and 100 for one minute, a normalized number of 100 for each would represent identical gameplay.

Another improvement might be reporting an average time to collect each object. In this way we can measure a patient's response time. Included would have to be a representation of average distance traveled per object. We leave these improvements for the next group to explore and implement.

Section 6) Meteors Game

The goal of Meteors is to collect as many stars as possible within the allotted time limit while avoiding the oncoming meteors. The player moves their hand left and right to move the rocket ship accordingly. The player is awarded one point for every star that they collect and is deducted one point for every meteor that they collide with. An encouraging prompt is displayed on the screen for every ten points that the player collects. The game is over once the timer runs out, at which point the game is redirected to a plot displaying the players entire game play history for Meteors.

Meteors has three different levels of difficulty. Modifying the difficulty will change two different parameters. The first parameter defines the number of meteors that can be on the display at any one time. This causes the spacing between each meteor to grow or shrink thus increasing or decreasing difficulty. Modifying the number of meteors that can be displayed at any one time also modifies the number of stars that are available at any one time. These differences can be seen in the two pictures screen shots presented below. The second parameter defines the amount of time it takes for a meteor to complete its trip from the top of the display to the bottom of the display. This causes the speed of the game to be increased or decreased thus changing the difficulty.

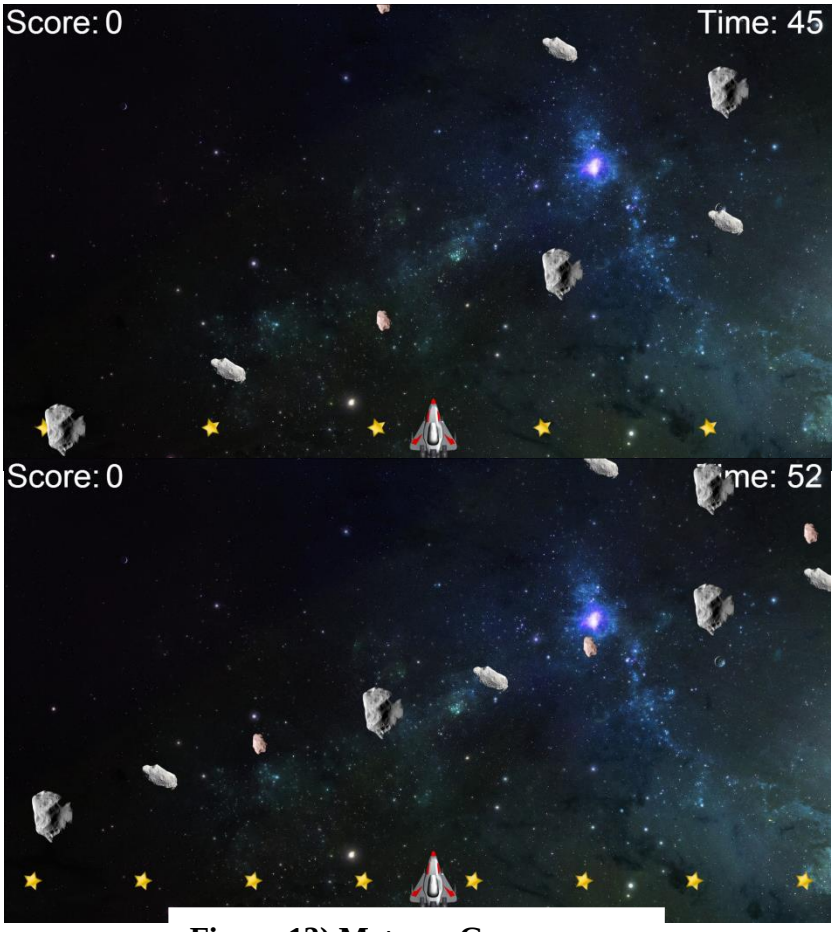

Figure 13) Meteors Game

Section 7) Pirates Cove

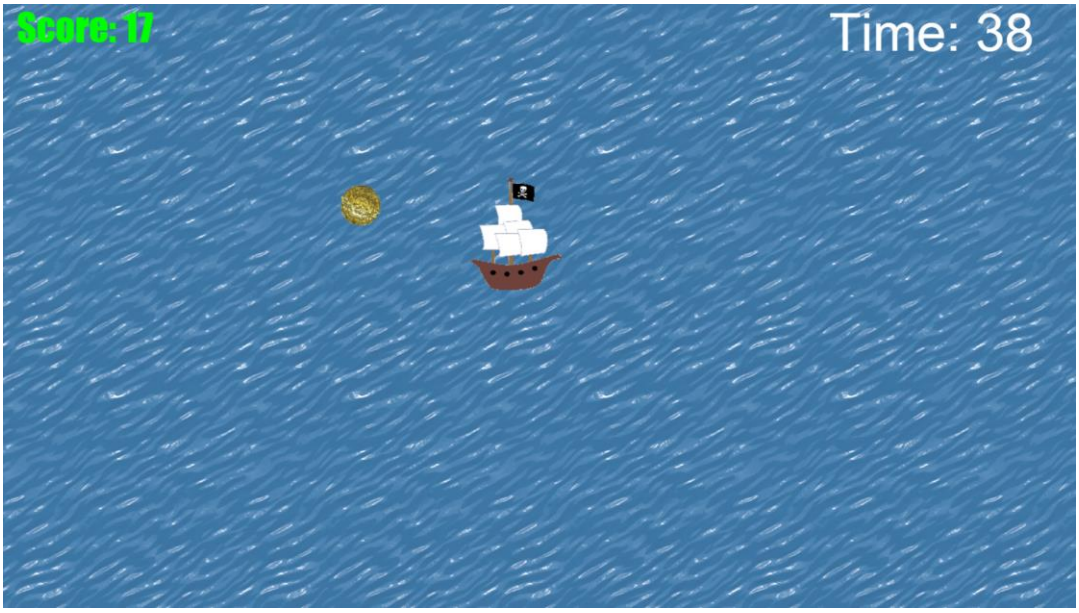

Figure 14) Pirates Cove

The Pirates Cove games casts the user as a pirate plundering the high seas for gold coins. The users hand is mapped to a pirate ship easily identified by the sails and Jolly Rodger flag. The coins are gold colored and have a Skull and Crossbones on the. Also visible is text displaying the score the user has achiever and the time remaining in the current session.

The idea behind this game was to develop a way to for the user to perform certain motion. We wanted to be able to place coins in predefined pattern. These patterns can be constructed in a shape we can conceive, offering a lot of flexibility. This particular game allows us to implement patterns in two dimensions. We isolated these motions into three categories, forward and backward motions, left and right motion and a combination of the two.

Forward motions were implemented by placing coins in gradual increments from the bottom of the screen to the top of the screen and back again. Left to right motions were implemented by placing coins in increments for the left to right side of the screen and back. Combinations of the two were done in a few different ways. The easiest is a diagonal criss-cross that creates an X onscreen. Increasingly difficult are the square, snake, circle and star patterns.

To reduce monotony we have included a random mode. This mode isn't as useful in rehab because of its lack of repeatability.

We designed each of the games to have varying levels of difficulty. In this game the difficulty affects the collision detection radius between the coin and ship. Originally we wanted to change the size of the coin to achieve the same results. Discussions with Matt, however, led us to believe that adjusting the collision detection radius and keeping the coin size the same was better to avoid confusing the patient's senses when adjusting difficulty.

The game also supports a sensitivity and time limit adjustment. Sensitivity is a scalar factor that multiplies the input from the Leap. Adjusting the scalar factor lets us specify how much the user icon will move in relation to the user's hand. The setting allows users with small ranges of motion to play. We can also use it to challenge users with larger ranges of motion by using a smaller scalar factor.

Section 8) Maze Game

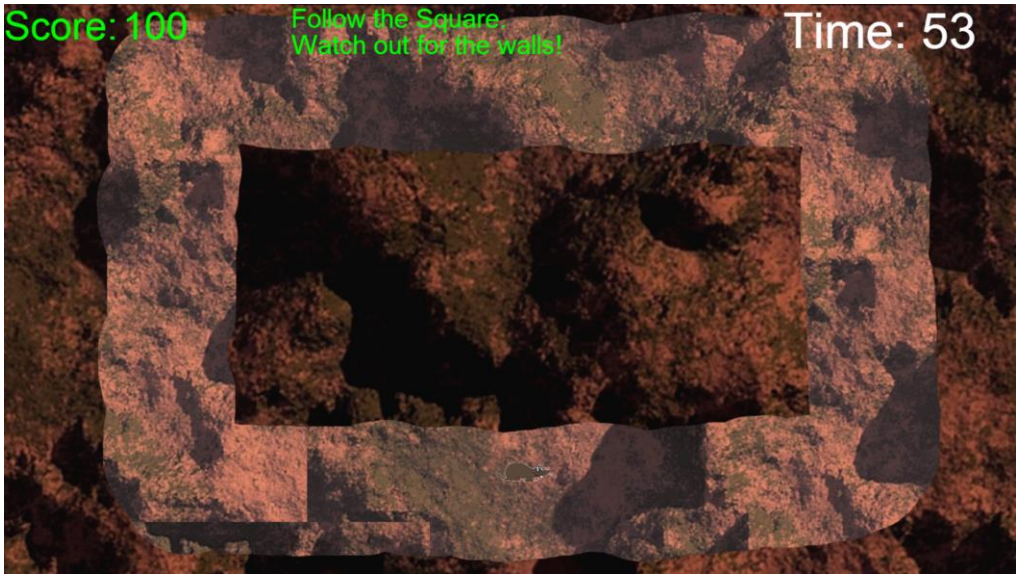

Figure 15) Maze Game

The Maze game casts the user as a mole traveling through tunnels underground. The patterns are completed in two ways. The first is by reaching opposite ends of the screen for the Snake, X, Up/Down, Left/Right patterns. The second is by completing a loop of the Circle and Square patterns.

Scoring is done slightly differently for the Maze game. For each collision with the boundary one point is removed from the total, beginning at 100. For each completion of the pattern the score is saved and then reset to 100. At the end of gameplay the score for all of the completions is averaged and reported to the database.

There are also a few pieces of information available on the game screen. In the top left corner is text indicating the score for this iteration. In the right is a display of how many seconds are remaining in the current session. Also displayed are directions telling the user which direction to move.

This game was born of the same initiative of the Treasure Game, but penalizes users who cannot follow the pattern accurately. The motivation behind this was to rehabilitate accuracy of movement. When reaching for a cup in the cupboard you don't want to know over everything else. We tried to recreate that situation with this game. Similar to the Treasure game we can rehab forward and backward, side to side and a combination of the two motions.

Forward and Backward motions were implemented by creating an image with a transparency through the middle from the bottom of the screen to the top of the screen. Left and right motions were implemented by drawing this transparency from the left side of the image to the right side of the image. More complicated images have also been drawn. These patterns require the user to move in a Square, Circle or Snake.

This game implements the different difficulty settings by loading images with transparencies of varying sizes. The larger path widths are easier to navigate, creating an easier difficulty setting. This game is probably the most difficult of the five we developed, requiring acute motor control to play accurately.

The game also supports a sensitivity and time limit adjustment. Sensitivity is a scalar factor that multiplies the input from the Leap. Adjusting the scalar factor lets us specify how much the user icon will move in relation to the user's hand. The setting allows users with small ranges of motion to play. We can also use it to challenge users with larger ranges of motion by using a smaller scalar factor.

Section 9) Water Drop

Figure 16) Water Drop

The game specifically targets left-to-right movement. It also helps patients who have a hard time keeping their hand stationary as once you get to the destination of the water drop, you should keep your hand in that position. Parameter that can be modified include: Sensitivity, Difficulty, and Time Limit. Other than sensitivity, and time limit which works the same for each game difficulty will change the rate of fall for the water drops.

The background images and music were meant to provide a calmer setting.

Section 10) Whack-A-Mole

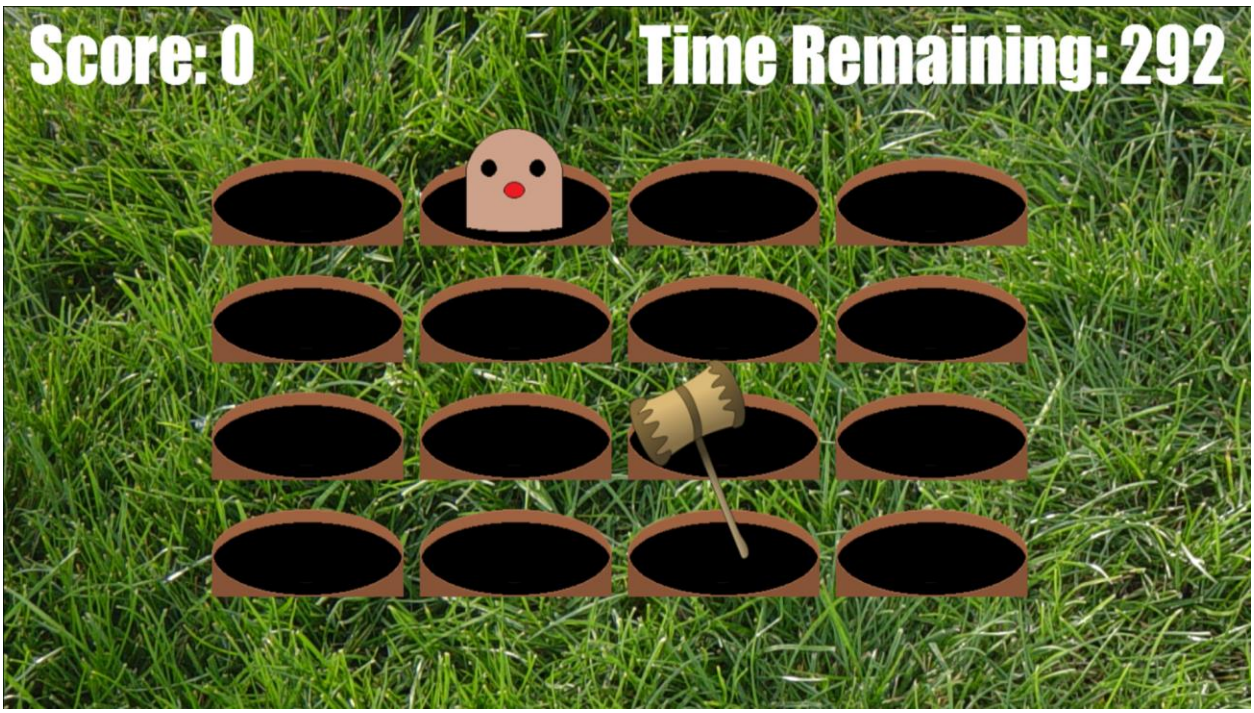

Figure 17) Whack-a-Mole

This game targets movement in both the X and Y axes, and can possibly be altered to include the Z axis in the future. The main strength of this game is that there are a wide range of patterns that can be implemented. Since there are nine mole holes you can adjust the patterns to go from left-to-right, up-and-down, diagonally or any combination of the three. It can also be implemented as a random pattern. The sensitivity and time setting work the same for each game while the difficulty setting adjusts the amount of time a mole will stay up out of its hole.

The background and music present a setting that it fun and outdoors, something of a carnival environment to provide a fun atmosphere.

Section 11) Database Improvements

During the second semester of our project we restructured a major chunk of our code. These code changes were performed for a number of reasons. First, we wanted to ensure that the system would be able to run without the need of an internet connection. Second, we needed a way to dynamically load game profile information whenever a user launched a game. Third, we needed to refine the structure of our database to make it somewhat simpler to maintain and expand upon. Finally, we wanted to consolidate certain aspects of our system into fewer files in order to make maintenance simpler.

All of the various web pages and games included in our system rely heavily on various Javascript libraries. These libraries include jQuery, LeapJS, the CreateJS suite, Highcharts, and SlickGrid. When our system was initially designed we were including these files from various external URLs. This means that the user needed to ensure that the computer that our system was running on had an internet connection. If an internet connection as not present our system would be rendered useless. To combat this issue we have consolidated all of the Javascript dependencies into our system in order to make them locally accessible. Not only does this remove the need for an internet connection, but it also helps to improve load times for our games since they are no longer required to retrieve remotely stored files.

The Web2Py web framework provides a nifty system called "Templating". Templating allows us to define the structure of a web page in standard HTML, but also insert chunks of Python logic into the HTML file. The Python logic will then perform manipulation on data or simply place a piece of data into the HTML. This allows us to define a single file that will dynamically adapt to any user. In order to streamline the retrieval of game profile information we decided to exploit the Web2Py templating system. The first step we took was merging all of the Javascript files for each game into their respective HTML files. This would allow use to insert game profile information directly into the Javascript itself via templating. We could have performed an AJAX request in order to retrieve the game profile information. However, we thought that a second request to the back-end code would be wasteful.

The first rendition of our database was rather obtuse and difficult to maintain so we chose to restructure it. We've removed the use of the "patients" and "therapists" table and replaced them with the "role" and "therapist" attributes located within the "person" table. The "role" attribute will define if a person is a therapist, patient, or unassigned. The "therapist" attribute will define which therapist this person is assigned to.

We chose to merge the code from the controllers of each of the games into the default Web2Py controller. We were experiencing issues with authentication whenever a user's game play information was being stored in the database. Each game now has two functions within the default controller. The first function handles retrieving a user's game profile information for the game and loading the game itself. The second function handles retrieving the game play information and inserting it into the database. Whenever a new game is added to the library the programmer only needs to implement these two functions to get the game to work properly.

Visual Progress Display

Our system needed an intuitive and easy to use way to view a patient's game play history. For our solution we decided to make use of the Highcharts and Slickgrid Javascript libraries.

Highcharts allows us to plot a user's game play history in a clean and interactive graph. From within this graph a user may show and hide a games plot, zoom in on a certain area, or export the graph to an image or PDF file. The user may hover their cursor over each point within the plot to show more detailed information about that data point.

Slickgrid allows us to present a user's game play history in a clean and easy to use data table. A user can sort this table by clicking on the header of the column that is to be sorted. A user may filter the table typing in their desired search filter into a search box. Finally, a user may double click on an entry within the table to show a Highcharts plot of the game play history for the selected game.

We have set up our games so that they will automatically redirect to a Highcharts plot upon completion. This plot will show all game play history for the completed game and will also include an entry for the game play session that had just ended. This is intended to give immediate feedback to the patient in hopes that they will see that they have improved from their previous plays.

Gameplay Settings

During our rendezvous with the OT department we were introduced to the "Just right challenge". The idea behind this is to make the games difficult enough so that they challenge the patient while not deterring the patient from continuing on with their therapy. In order to meet this goal we implemented a game profile system. This system will allow a therapist to tweak various settings in order to tune each game to the needs of each patient.

Our first task for creating a game profile system was to define the various settings that would be included. We decided to provide difficulty, time duration, sensitivity, and pattern settings. The behavior of each of these settings is outlined below:

Difficulty:

Meteors: Modifying the difficulty will cause the spacing between each meteor to be wider in lower difficulties and narrower in higher difficulties.

Maze: Modifying the difficulty will cause the path to be wider in lower difficulties and narrower in higher difficulties.

Whack-A-Mole: Modifying the difficulty will increase the duration that each mole will stay up in lower difficulties and decrease it in higher difficulties

Pirates Cove: Modifying the difficulty will increase the collision area around a coin in lower difficulties and decrease it in higher difficulties.

Water Drops: Modifying the difficulty will decrease the falling speed of the water drops in lower difficulties and increase it in higher difficulties.

Time: Modifying the time will determine the duration of game play. The game must run for this duration in order for the score to be recorded.
Sensitivity: Sensitivity defines the correlation between movement of the players hand and the movement of the player's character on the display. A lower requires the player to move their hand further in order to move their game character some distance. A higher sensitivity will allow the user to move their hand less to achieve the same movement of their game character.
Pattern: Depending on the option selected the user will be required to repeat some sort of patter during gameplay. For example, selecting "Square" will make the user repeatedly move their game character around the game play area in a square pattern.

In order to store these settings we needed to create a database table. This table consists of all game profiles for all users of the system. To retrieve a particular game profile from this table the developer should query for the currently logged in user's ID and the title of the desired game profile.

Therapist Profiles

In order to provide a simply way to administrate all the users of our system we created a specific view for the therapists. This view is similar to that of a patient. However, this view contains two extra tabs along the top of the main window: "Administration" and "My Patients". The "Administration" tab will allow the therapist to view all users who are currently registered in the system. The "My Patients" behaves similarly to the "Administration" tab except for it will only display a list of users who have been assigned to the currently logged in therapist.

From within either the "Administration" or "My Patients" tab the therapist can launch the user administration tool. This tool will allow the user to modify and view various information regarding a particular user including profile information, game play history, and game profiles.

The administration tool allows a therapist to modify a user's profile information such as user name, email address, and password. The therapist may also enter in extra information into the "Other Information" text area. The "Other Information" text area is not viewable be a patient and is intended to be used for quick notes regarding a patient's therapy.

A user's game play history can be viewed from within the user administration tool by clicking on the "Game Play History" tab. Within this view a therapist may plot the scores of all games that have been played by a particular user in a Highcharts line graph. This graph allows a therapist to hide and show various plots, zoom in on certain portions, and export a plot to an image or pdf file. A user's game play history may also be viewed in a table which will allow the therapist to easily sort and search through the data.

Finally, a therapist may modify a user's game profile information from within the user administration tool. This will allow a therapist to change the difficulty, duration of game play, sensitivity, and pattern of each game.

Chapter III – Alternatives, Testing and Ethics

Alternative Approaches

There have been two previous iterations of this project that explored other solutions. The first project done by Jacob Poore and Baris Tevfik explored the use of Web Cams to implement augmented reality games. These games used 3D Graphics and Augmented Reality tracking libraries to achieve in home rehabilitation results. Overall they were successful, but Professor Pasricha wanted to explore mobile solutions[4]. Alex Vlahinos, Gregorio Campuzano, and Nathaniel Olsen explored implementing these games on an iPad. While successful, the iPad lacks the processing power to perform complex object recognition, severely limiting the games that could be implemented.

As a result of these findings, the decision was made to shift back to personal computer based systems. Because webcams had already been explored, this project chose to explore the use of hardware built with gaming in mind. The alternative to the Leap Motion Controller was the Microsoft Kinect. The Microsoft Kinect is used in conjunction with the xBox 360 to play user interactive games like Kinect Sports or Dance Central. Playing games with the Microsoft Kinect on your computer is a much more difficult process. It requires the installation of Microsoft Visual as well as the requisite drivers, a task we deemed too complex for a patient.

We also considered using alternate languages for developing our games. When still considering the Microsoft Connect languages like C++ and C# were considered. The Microsoft Kinect was optimized for these languages, especially C#. Once the decision was made to abandon the Microsoft Kinect it re-opened other programming languages. We did not explore other development routes for the implementation of our database architecture. We went directly to the HTML and web2py framework.

Testing

The extent of our testing in the first semester was to find feasible solutions to implement into our system framework. We tested our hardware to decide between the Microsoft Kinect and Leap Motion Controller. Once we decided on the Leap Motion Controller we tested the capabilities and accuracy of the Leap Motion Controller. Following this, we needed to test the Leap Motion Controller to ensure the incorporation of real world object would not throw off the sensor. We also needed to test that environmental variables would not throw off the sensor.

Testing to decide between the Leap Motion Controller and Microsoft Kinect consisted of exploring the capabilities of both. We played many of the available development library games from Kinect Development Library and modified the code to get a feel for the ease of programming the device. The Leap Motion Controller is a very new device however and barely had a development community. The testing of this device included playing some of the available apps and programming a few of our own games with the available Leap API.

Testing of the Leap Motion Controllers range was rather simple. We needed to guarantee it provided sufficient area for the user to move their hand. According to the Leap Website the device has around 8 ft^3 of detection area. For reasons mentioned earlier the Leap Motion Controller needed to be top mounted. We adjusted the height at which it was mounted and found that 12" above the users hand provided the most consistent data. Furthermore, the user still experienced just over two feet of motion available to them.

Following this discovery we needed to categorize the objects that could be used with the Leap Motion Controller. Extensive testing has not been completed on this, but we know that objects too tall interfere with the Leap's sensors. Color does not have an impact, but translucency does seem to. Solid objects are better for our purposes. Room lighting has no effect, and neither does the skin color of the patient. During our testing we found dark backgrounds worked better with the leap. These dark backgrounds reflected less infrared light and interfered with the Leap's sensors less.

In the second semester our testing focus shifted to refinement of the games. Through working with the occupational therapy department we were able to get a better idea of what they need from the games.

The first form of feedback was difficulty settings for each game. Matt introduced us to the concept of the just right challenge. This task requires configurability for each game, and is our reasoning behind including sensitivity and difficulty settings.

The OT department also requested visual and audio feedback be integrated into the system. Audio feedback was done in the form of playing sounds when the target action was completed. For example a coin dropping into a pile of coins is played every time the user collects a coin in the pirate game. Visual feedback happens in two ways. The first way we implemented this form of feedback is through the use of in game messages that give positive, encouraging support. The second form of visual feedback is through the use of graphing libraries to display results once they complete the game. These graphs will show the patient (and the therapist) how they have done compared to previous sessions.

We also had the opportunity to work with two stroke patients, Debbie and Craig, and receive their input on the project. This testing went as well as we could have hoped. Debbie and Craig both said the system was fun and different from any rehab they had previously done. Debbie even discovered she had more use of her arm than he previously though. By resting her hand on the table she was able to extend her arm in front of her, a step toward rehabbing those muscles.

We also found we had a few things to work on. We ran into an issue with the website not loading the results immediately following the conclusion of gameplay. Debbie also requested we take into account ergonomics. She suggested this could be done in two ways. Resting her hand on the table meant that her arm would rub against the table. Stroke patients may also experience a lack of muscle mass due to lack of use of their arm. In this case the bone is directly striking the table, causing pain. The second suggestion had to do with ergonomic objects to use in gameplay. Patients sometimes suffer from muscle paralysis after a stroke, which makes holding large

diameter objects difficult. This is something we had not anticipated. Solutions to this problem would include ergonomic-minded objects, including support for the wrist to keep it from curling. Objects of varying diameter would assist with this problem. Communication with experts in the area would be necessary to develop a solution.

Ethical Concerns

Our biggest concern with this project is copyright infringement. We are leveraging many aspects of our project from previous work or examples. We must be careful not to steal anyone's work, as it would put the University at risk. For example, the three of us are not graphic designers. As a result, the three of us have built the theme of our games using images from the internet. These images must be open source images, and if we cannot find open source images to suit our needs we must construct images of our own.

Furthermore, we must be careful to avoid copying code. Any examples we encounter on the internet need to be created with the intent of teaching, and cannot be copyright material. Again, using this code will put us and the University at risk.

Chapter IV) Clinical Trials

Part I) Craig

After working with Matt to bring the system to his standards we had the opportunity to perform some initial clinical trials with our project. The idea behind this was to bring in a few people who are in different stages of their rehab and get their feedback.

Our areas of focus in these trials were different for each user. For Craig we were interested in the adaptability of each game. We wanted to know if the difficulty was too hard or too easy. To do this we had him play each game three times for one minute each session. The first session would be played on easy, the second on medium and the last on hard. We kept the pattern and sensitivity the same.

Figure 18) Craig Playing Whack-a-Mole

We had no numerical system to determine the proper difficulty. This was each patients first time playing, and as such we had no idea of their skill level. Instead we relied mostly on feel and patient feedback to determine if we could find a difficulty that suited Craig. Important to

consider was the small status of Craig's stroke, which allowed him greater proficiency of gameplay.

Craig had no trouble playing any of the games. We noticed two overall trends in Craig's scores. The first was an acclimation increase as a result of getting accustomed to the Leap and the game. This typically happened between the easy and medium difficulty settings. The second trend was a leveling out or reduction between the medium and hard difficulty settings. We took this as an indication that we had found a setting that challenged Craig.

The fifth dataset contains six points. The last three were played with identical settings. The goal of this was to determine if Craig could achieve a score improvement. A score improvement indicates Craig's comfort with the system and motor control are improving.

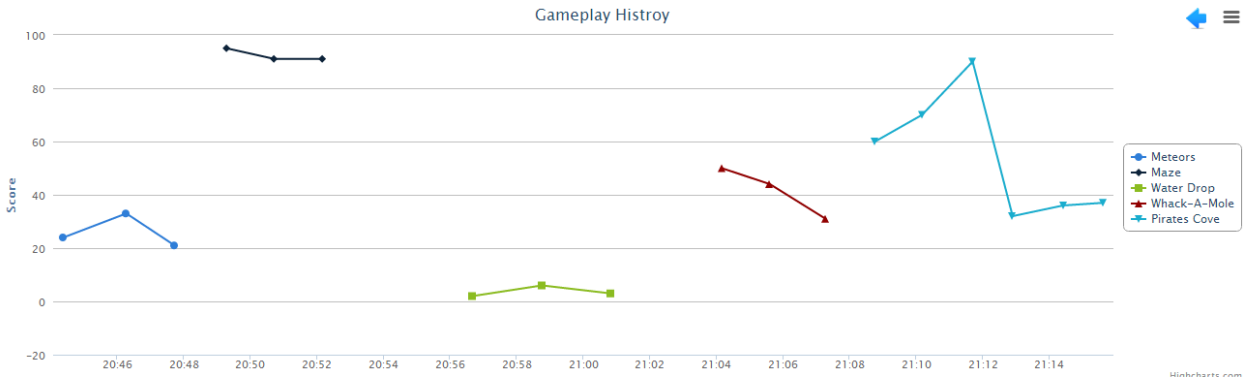

Figure 19) Craig's Gameplay History

Part II) Debbie

Debbie's condition was much more severe. As a result we made the decision to test for improvement in gameplay through multiple repetitions. We played three Games with her, Water Drops, Whack-a-Mole and Pirates Cove. For each game we played on a difficulty setting of easy, a medium sensitivity and a one minute duration. For the Whack-a-Mole and Pirates game we loaded a forward and backward pattern for reasons I will discuss later.

Figure 20) Debbie playing Pirates Cove

We began by playing the Water drops game. Debbie nailed this game, collecting every available water drop after getting accustomed to the system. Finding this too easy, we decided to move on. Debbie had expressed an inability to perform extensions of her arm. Having expressed this, we decided to push Debbie a little bit by loading an Up/Down pattern into the Pirates Cove and Whack-a-Mole games.

Due to a scoring bug in the Pirates Cove game we only analyzed data from the Whack-a-Mole game. Debbie's scores remained relatively flat, but her elation at being able to extend her arm forward was the real success. Debbie doubted she could perform this motion until playing these games.

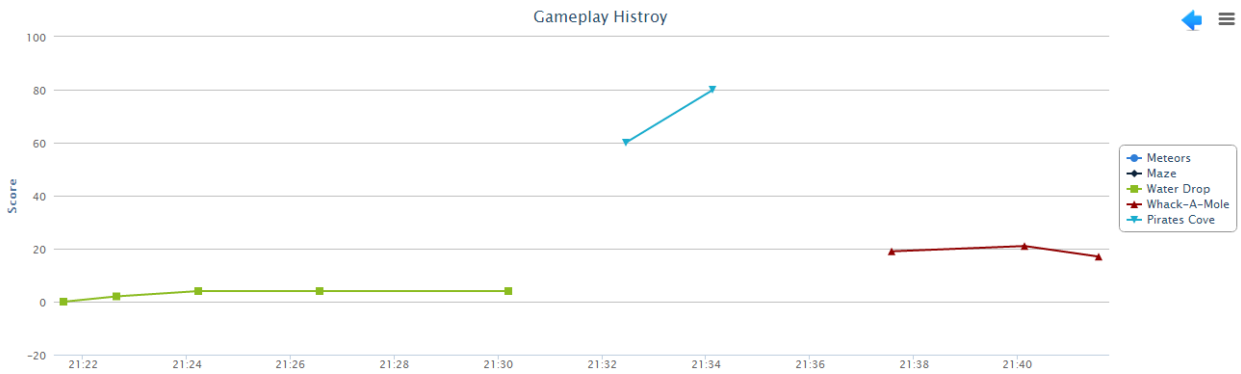

Figure 21) Debbie's Gameplay History

Part III) Observations

While playing we made a few key observations. The first was the level of difficulty of the Water Drops games. The drops fell far too slowly and infrequently to challenge the patient. The Hard setting was more suited to an easy/medium setting.

The coin game had an unreliable scoring system, awarding point for completion of a pattern, rather than per coin. This resulted in poor granularity for determining performance.

The Whack-a-Mole "Whack" sound was inaudible. This resulted in poor positive feedback.

The ergonomic setup of our system was something we did not consider. Debbie reported pain in her arm from rubbing on the edge of the Table. We temporarily solved this by placing a sweater under her arm. Debbie also had trouble gripping the object we had provided due to the muscle paralysis in her hand. She requested we either Velcro the object to herd hand (not ideal) or provide objects that are ergonomically designed. Ergonomic object would be ideal because we want patients to be able to grip the object under their own power, rather than attaching them to their hand. This way they rehabilitate muscles used to hold objects.

Chapter V - Conclusion

All features discussed in this report have been implemented and demonstrated to work. The most important feature of our system is its adaptability. New games can be developed efficiently to isolate new motions, incorporate new rehabilitation techniques and improve the quality of the system. Our database has the capability to handle varying numbers of user and the data demands associated with each user.

The Leap recognizes motion in three dimensions, giving us the capability to build a variety of games. Games can be built to isolate motion in any combination of these directions. For our purposes we focused on stimulating motion in the X and Z dimension. We reasoned that most user would struggle holding their hand in midair, and gave them the benefit of a table to rest their hands on. This eliminates up and down motion but increases the feasibility of our system.

Having proved the functionality of our system, the remaining work lies in streamlining it. In no particular order, the games need to be standardized to meet the following requirements. All of the games need a game menu to configure the games to a user's needs. All games need to be adjustable to the user's windows size. Upon adjustment of the window the game needs to respond accordingly.

The database coordinates information passing between the therapist and his patients. Therapists can control the games available to a patient, the required game time and the difficulty. Furthermore, patient progression data is only available to the patient's therapist and the patient himself. The database can have any number of patient to therapist groupings and therapist sub groups. This organization is an effort to keep patient confidentiality between the patient and their therapist.

The database supports two separate types of profiles that distinguish between therapists and patients. Patient profiles are restricted and allow for the user to view results, access personal information and play games. The therapist profile has extended capabilities. A therapist can view each individual patient's information and results, as well as adjust the setting for each patients games.

Our games are fully configurable to meet the patient's needs. We have designed each game to receive inputs from the therapist about the difficulty and length of gameplay. We have also implemented the ability for the therapist to choose between a random object placement mode and from a library of pre-defined pattern in which objects are placed. These libraries allow for the isolation of specific motions.

We had the opportunity to have to patients who had done previous research with the CSU OT department work with our project and give us feedback. This testing was extremely productive because the system was both well-received and the participants had good constructive criticism.

Future Work

We have built a system that is highly scale able, meaning new pieces of hardware can be integrated into the games. For example if a user needs to develop lower body strength, games could be developed for the Microsoft Kinect (or another apparatus). When this particular patient logs in, a different set of games from the Leap library would be loaded.

Current plans are for the system to be expanded to include Google Glass. Google glass is a head mounted display that allows users to easily use software like Gmail and Google maps. Students will have to explore how this product can be adapted to a rehabilitation or assistive framework.

There is also room for improvement with regards to our work. As discussed earlier, ergonomics should be taken into account. Muscle paralysis can happen following a stroke, and a solution should be developed to assist people who cannot properly hold objects. The rehabilitation of grip muscles is essential to leading a normal life, and if patients cannot do this incremental steps must be provided.

The OT department also requested an assignment feature. This would allow a therapist to select a user and assign them a combination of games to play. These combinations could include instances of the same game with differing setting or different games.

Patients also tend to make compensatory movements with their bodies when shoulder or arm muscles are impaired. Ideally the games we have should be played with upper arm and shoulder muscles only. A system to encourage this could be developed.

Advances in technology have seen the evolution of biometric sensors. The Biometric sensors we would be interested in deal with muscle activation and usage. More information about how a patient is using the muscles in their body can assist a therapist in making assessment of progress from a remote location.

As discussed in Section 5, improvements in performance metrics are necessary to accurately report the status of the user. These improvements may include score per time and average collection time numbers as mentioned earlier, but are certainly not limited to these.

REFERENCES

[1] History of Physical Medicine and Rehabilitation. *Medical College of Wisconsin. [online] 12 (4) 2012.* http://www.mcw.edu/physicalmedicine/history.htm (Accessed: 9 December 2013)

[2] Same Gelman, Nalini Vishnoi, Zoran Duric, Naomi Lynn Gerber. "Validating the accuracy of microsoft kinect depth information."

[3] Postage Price Calculator. *United States Postal Service.* [online] http://postcalc.usps.com/ (Accessed: 11 December 2013)

[4] Augmented Reality Games for Neurological Rehabilitation. *Colorado State University.* [online] 5 (4) 2012. http://www.engr.colostate.edu/ece-sr-design/AY11/rehabilitation/summary.htm (Accessed: 10 December 2013)

APPENDICES

- Appendix A

Augmented Reality - AR
Cerebral Palsy - CP
Graphic User Interface – GUI
Traumatic Brain Injury - TBI

- Appendix B

The cost of our project has been minimal. Using open source software and Academic Software downloads available through sites like Dreamspark we have been able to save most of our money. The other costs incurred by our project have come out of team members pockets. Below is a summary of available funds and purchases made with them. Currently our only future plans are for a large display device to increase the visibility of our project. We envision a setup with a viewing device moved back from the user, the laptop off to the side (its only purpose is processing now) and only the Leap Motion Controller in the immediate area.

Beginning of the Project Total	$374.73
Josh Leap Motion (10/03)	-$90
Updated Total	$284.73

Table 1: Team Expenditures

Costs incurred to individual team members include two Leap Motion Controllers and parts for the demo mount. These Leap Motion Controllers were purchased with the intent of keeping them once the project had concluded or been handed off. As such, it was deemed improper to use projects funds for them.

Ryan’s Leap Motion	-$90.00
Jake’s Leap Motion	-$90.00
Jake Mount Hardware	-$10.00
Computer Matt	-$10.00

Table 2: Personal Expenditures

The only aspect of our project that has any effect on shipping manufacturing costs is the Leap Motion Controller mount. While it is not necessary to play any of the games, it assists

people with an impairment of the upper body by allowing them to rest their arm on a table. The solution we currently have is crude and meant to demonstrate the concept. We envision future development costs to create a better solution.

This is not our area of expertise however, and we have not realistic estimates of the cost for labor, shop, mechanical hours and parts. We can tell you our current solution costs under ten dollars and takes around half an hour to assemble. Shipping our current solution would cost $13 under a USPS flat rate shipping system[3]. There are no heat sensitivity concerns to ship our current mount.

- Appendix C

Our project was difficult to categorize as a breakdown of individual tasks. As a result we created a very general description of milestones we hoped to hit. In hindsight we wish we would have come up with a better way to illustrate the progress we had made. This outline would be very good from a corporate perspective, it is difficult to track microscopic success from a teap perspective.

Revised September 30th

Phase 1 – Investigate capabilities of Hardware - Delivery date: September 20th

Leap Motion Controller – Jacob Thornton / Ryan Nash:

Investigate the various positions that the leap motion can be placed in and how well the device can track the user. We will test the Leap Motion by placing it above, below, and to the sides of the viewing area. We will also investigate how accurate the device is at tracking a user's movement. We will also investigate the type of data that is being gathered by the Leap Motion Controller and how to best use this data to create games for physical rehabilitation exercises.

Kinect – Joshua Wallace / Ryan Nash:

Investigate how accurately the device can track user movement, skeletal matching, speech integration, and object manipulation. Research previous projects that have used the Kinect as a human interface device on a PC and apply what we learn to our programs.

Phase 2 – Simple Game Development – Delivery date: November 30th – All team members

Simple games will be developed on both pieces of hardware in order to gain knowledge of how each piece of hardware works and how to best write software for each.

Jacob Thornton

Jacob Thornton will focus on the Leap Motion Controller. Games that will be created by Jacob will include brick breaker, follow the leader, and car racing. These games will focus on fine grain movement of the arms and hands in the X, Y, and Z axis. Jacob will be writing software in Python, Javascript, and Java.

October 11th – Have rough design of graphics for games created

October 25th – Have general structure of games implemented and at least one game ready for testing.

November 6th – Have at least two games ready for testing. One of which has already been revised.

November 20th – Have all three games completed and in testing. Revisions applied to at least two games.

November 30th – All three games completed and revised.

Joshua Wallace

Joshua Wallace will focus on the Kinect. Games that will be created by Joshua will include pop the balloon and Simon says. These games will focus on a broader range of movement as opposed to the leap motion controller. Movement will be primarily on the X and Y axis. Joshua will be writing software in C# and possibly C++ if the need arises.

October 11th – detailed outline and layout of Simon says and balloon game including list of features, objectives and movements.

October 25th – Basic functionality including movement tracking and general game structure.

November 6th – Have at least one game ready for testing.

November 20th – Have one game completed and the other ready for testing.

November 30th – Two games completed.

Ryan Nash

Ryan Nash will work on both the Kinect and Leap Motion Controller. Ryan will work primarily on the leap, but assist Josh where applicable. Ryan will focus on games target toward simulating real world actions taken for granted by most people. Ryan will be programming software for the Kinect in C# and software for the Leap Motion Controller in Javascript.

October 2nd – Meet with Prof. Pasricha to brainstorm ides for action recreation games.

October 11th – Execution plan delivered.

October 25th – Game engine implemented.

November 6th – Games finalized and reviewed by team.

November 13th – Critiques implemented and shown to Prof. Pasricha.

November 30th – Investigate new possibilities for recreation in leap and determine course of action.

Phase 3 – Refinement – Delivery Date: December 19th – All team members

After some simple games have been created we will re-work them in order to better suit the ultimate goal of the game. This may include enhancing graphics, enhancing response time, scoring, web access, menus, voice recognition, and sounds. New revisions may be thought of as we progress through the project and gather results and feedback from clinical trials.

Semester Two

Phase 4 – Clinical Trials – Delivery Date: Pending Meeting With Sudeep – All team members

Games that have been developed during phase 2 will be tested with patients. Feedback will be gathered and will be used to further improve the capabilities and functionalities of the games.

Course of action with be determined by outcome of these trials.

Appendix D

ACKNOWLEDGMENTS

We would like to thank Sudeep Pasricha for his assistance in the development of our project. His support has been invaluable in developing the Augmented Reality Games system.

We would also like to thank Matthew Malcom for his input developing rehabilitation oriented tasks.

SOURCE CODE REMOVED

For information about source code
Contact Advisor Sudeep Pasricha